

DOI: 10.17058/rjp.v5i2.5770

META-HEURÍSTICAS NA CONFIGURAÇÃO DE PARÂMETROS EM SIMULAÇÃO COMPUTACIONAL

*Guilherme Tonini Botassoli¹**Rafael Alvise Alberti²**João Carlos Furtado³**Liane Mahlmann Kipper⁴*

RESUMO

Nesta pesquisa, o objetivo é desenvolver uma ferramenta que visa otimizar processos organizacionais através de algoritmos inteligentes, conhecidos como meta-heurísticas. Para o processo de otimização, o algoritmo altera parâmetros do modelo de processos em estudo de forma estratégica e analisa a resposta a essa mudança. Para teste, é utilizado o modelo de um centro cirúrgico real onde serão analisadas variáveis como tempo de espera dos pacientes e ocupação dos recursos. A cadeia de processos do modelo se estende desde a chegada dos pacientes na sala de espera até a sua alta e liberação. Ao final, o algoritmo gera um relatório com as melhores configurações encontradas no decorrer de sua execução e implanta no modelo a configuração 'ótima'. Todo o processo é feito dentro do Software Rockwell Arena, desde o desenvolvimento do modelo do centro cirúrgico para simulação até a implementação e execução do algoritmo. Os resultados obtidos foram satisfatórios; tem-se, por exemplo, uma redução de 90% no maior tempo de espera registrada entre os pacientes para os dois primeiros casos simulados.

Palavras-chave: Simulação de processos. Otimização de processos. Meta-heurísticas. Gestão hospitalar.

ABSTRACT

This research aims to develop a tool to optimize business processes through intelligent algorithms known as meta-heuristics. For the optimization process, the algorithm strategically changes the parameters of the process model in study and analyzes the response to this change. For testing, we use the model of a real operating room where it is analyzed variables such as patients' waiting time and resources' occupation. The process chain of the model extends from the arrival of patients in the waiting room until their release. At the end, the algorithm generates a report with the best settings found and it deploys the 'great' optimal configuration in the model. The whole process is done within the Rockwell Arena software, from the development of the operating room simulation model to the implementation and execution of the algorithm. The results were satisfactory; we had, for example, a 90% reduction in the longest waiting time recorded among patients for the first two simulated cases.

¹ Aluno do Curso de Engenharia Elétrica da Universidade de Santa Cruz do Sul. <guibotass@live.com>

² Professor dos Cursos de Engenharia de Produção e Engenharia Civil da Faculdade de Itapiranga - FAI, Mestre em Sistemas e Processos Industriais da Universidade de Santa Cruz do Sul - UNISC.

³ Professor do Departamento de Sistemas e Processos Industriais da Universidade de Santa Cruz do Sul - UNISC. <jcarlosf@unisc.br>

⁴ Professora do Departamento de Sistemas e Processos Industriais da Universidade de Santa Cruz do Sul - UNISC. <liane@unisc.br>

Keywords: Process Simulation. Process Optimization. Meta-heuristics. Hospital management.

1. INTRODUÇÃO

O hospital é, certamente, a forma mais complexa de organização humana que nos propomos a administrar. Sua complexidade inicia-se pela natureza de sua missão. Tecnicamente, um hospital é um prestador de serviços, mas é grande a distância entre salvar vidas e as outras atividades do setor de serviços (SOUZA, CARVALHO E LIBOREIRO, 2006). Machado *et al.* (2014) afirmam que essas organizações que prestam serviços de saúde são desafiadas a buscar modelos consistentes e integrados que possam aumentar o resultado máximo e evitar o desperdício de recursos próprios ou do governo, que são escassos.

Dentro do hospital, uma das áreas funcionais mais críticas é o centro cirúrgico. Tal setor move quase 70% das admissões do hospital e determina a maior parte de seus custos. Melhorar o seu desempenho, portanto, representa um objetivo estratégico para um número crescente de hospitais (CAPPANERA, VISINTIN e BANDITORI, 2014). Portanto, o centro cirúrgico é o local onde será aplicado este estudo.

Nesta pesquisa o objetivo é desenvolver uma ferramenta que visa otimizar processos organizacionais através de algoritmos inteligentes, conhecidos como meta-heurísticas. Para tal, serão exploradas funcionalidades do software Rockwell Arena (2015), software este muito utilizado na gestão para simulação de processos (modelagem, controle de recursos, entre outros). Ele também possibilita a integração com programação através de um ambiente VBA (programação na linguagem *Visual Basic for Applications*), como ocorre no Microsoft Excel, por exemplo. Dessa forma, um algoritmo foi desenvolvido nesse ambiente para controlar e otimizar o modelo de processos desenvolvido no ambiente principal do software.

Para o processo de otimização, o algoritmo altera parâmetros do modelo de processos em estudo de forma estratégica (utilizando os conceitos das meta-heurísticas) e analisa a resposta a essa mudança. Esta rotina se repete até que seja alcançado algum critério pré-estabelecido. Exemplos de respostas a serem analisadas são a produção, ociosidade de recursos, tempos de espera, filas, custos, entre outros.

O modelo desenvolvido é baseado na realidade do centro cirúrgico do Hospital Universitário da PUCPR. A cadeia de processos estende-se desde a chegada dos pacientes na sala de espera até a sua alta e liberação. As respostas a serem analisadas são o tempo de espera dos pacientes e a ocupação dos recursos (salas de cirurgia, de alta, enfermeiros e

outros). Ao final, o algoritmo gera um relatório com as melhores configurações encontradas no decorrer de sua execução e implanta no modelo a configuração ‘ótima’.

O artigo está dividido em quatro partes. Na primeira, introduz-se conceitos sobre simulação, otimização, o método enxame de partículas e algoritmos genéticos. Depois mostra-se a metodologia. Em resultados apresenta-se o modelo para simulação desenvolvido, o algoritmo, a resposta a ser analisada pelo algoritmo após o processo de simulação, e o centro cirúrgico. Ainda em resultados simula-se um aumento de demanda por cirurgias e ampliação dos recursos necessária para eliminar filas. Por fim expõe-se as conclusões.

2. FUNDAMENTAÇÃO TEÓRICA

A seguir apresenta-se uma breve revisão bibliográfica sobre simulação e otimização, e sobre as meta-heurísticas utilizadas neste trabalho, o método enxame de partículas e algoritmos genéticos.

2.1 Simulação e otimização

Carvalho (2003) define, de forma pragmática, o processo de modelagem e simulação como sendo uma experimentação computacional, onde são usados modelos de um sistema real ou idealizado para o estudo de problemas reais de natureza complexa. Desta forma, é possível testar diferentes alternativas operacionais, a fim de encontrar e propor melhores formas de operação que visem a otimização do sistema como um todo.

Para Costa (2011), simulação é a importação da realidade para um ambiente controlado onde se pode estudar o comportamento do mesmo, sob diversas condições, sem riscos físicos e/ou grandes custos envolvidos.

No processo de modelagem, usando o simulador Arena, o modelo é estruturado e codificado com base na linguagem de simulação SIMAN por meio da seleção dos módulos que contêm as características dos processos a serem modelados. O modelador não precisa necessariamente conhecer esta linguagem para construir um modelo, pois, utilizando os modelos disponibilizados pelo Arena, o usuário pode extrair módulos, posicioná-los e parametrizá-los de acordo com as características observadas no sistema real (SAKURADA e MIYAKE, 2009).

Embora a simulação computacional apresente vantagens, como as apresentadas acima, ela possui algumas limitações, pois se trata de uma ferramenta avaliadora e não geradora de soluções. Uma abordagem que faz com que esta situação comece a ser mudada é

a utilização de técnicas de otimização na simulação computacional (CHWIF e MEDINA, 2006).

Esmin (2005) define a otimização como sendo a tarefa de determinar as “melhores” soluções para certos problemas matematicamente formulados. Toda tarefa de busca e otimização possui vários componentes, entre eles: o espaço de busca, onde são consideradas todas as possibilidades de solução de um determinado problema, e a função de avaliação, que é uma maneira de avaliar os membros do espaço de busca.

Chwif e Medina (2006), definindo a relação simulação/otimização, dizem que, enquanto a simulação possibilita encontrar gargalos na linha de produção, a otimização busca melhores configurações ou rearranjos na mesma. A prática da utilização de ferramentas de otimização em simuladores já é observada a algum tempo na literatura e este trabalho visa, entre outras coisas, contribuir para esta área de estudo.

2.2 Método Enxame de Partículas

Para Epitropakis, Plagianakos e Vrahatis (2012), o Método Enxame de Partículas (em inglês, *Particle Swarm Optimization*, PSO) é um método de Computação Evolucionário que pertence à grande classe de métodos de Enxame Inteligentes. O PSO foi introduzido por Kennedy e Eberhart (1995) e é inspirado pelo comportamento social de um bando de pássaros e um cardume de peixes. É baseado em um modelo psicossocial de influência e aprendizagem. As principais hipóteses para o desenvolvimento do PSO dizem que o compartilhamento de informações entre membros de uma mesma espécie proporciona vantagem à espécie.

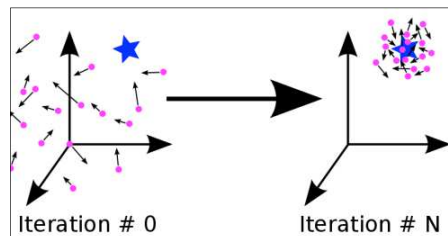
Jiang, Li e Huang (2013) definem o PSO como um algoritmo estocástico que funciona com base em uma população de partículas (que são possíveis soluções) para resolver problemas de otimização. Essas partículas iniciam-se em lugares aleatórios e procuram pela solução “ótima” em um espaço limitado, atualizando sua velocidade e posição continuamente. O algoritmo termina quando o critério de parada, definido pelo usuário, é alcançado.

J.A. Lazzús e M. Rivera (2013) complementam explicando que, em um sistema de PSO, cada partícula ajusta sua posição no espaço de pesquisa, de acordo com sua própria experiência e na experiência das partículas vizinhas. Logo, a partícula faz uso da melhor posição encontrada por ela mesma e a melhor posição encontrada entre todas as partículas para reajustar sua direção (figura 1) em busca da solução ótima.

Define-se solução ótima, no PSO, como a melhor posição possível para as partículas dentro do espaço pré-definido, ou seja, o resultado ideal. Após a execução do algoritmo, a

solução ótima pode ter sido alcançada ou não. Nos casos onde ela não é encontrada os possíveis motivos são, entre outros, número insuficiente de partículas ou de troca de informações, complexidade do problema e tamanho do espaço de busca.

Figura 1: Resultado após N iterações - as partículas chegam ao resultado satisfatório.



Fonte: BECKER, J. An Introduction to Particle Swarm Optimization (PSO) with Applications to Antenna Optimization Problems. *Post*, 2013. Disponível em: <<http://wirelesstechthoughts.blogspot.com.br/2013/06/an-introduction-to-particle-swarm.html>>. Acessado em: 04/03/2015.

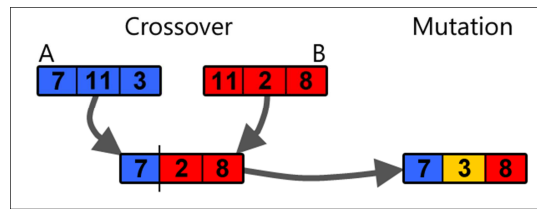
Epitropakis, Plagianakos e Vrahatis (2012) ressaltam vantagens do método, mostrando que o mesmo é capaz de trabalhar com funções-objetivo não-diferenciáveis, descontínuas e multimodais. Além disso, este método tem ganhado crescente popularidade nos últimos anos devido a sua relativa simplicidade e capacidade de lidar com aplicações reais de forma eficiente e eficaz.

2.3 Algoritmos Genéticos

Sherer *et al.* (2012) introduz Algoritmos Genéticos (Genetic Algorithms, GA) como uma classe de técnicas de pesquisa global normalmente utilizada em engenharia e ciência para estimação de parâmetros multi-variáveis; esta ferramenta tem sido aplicada com sucesso em problemas como distribuição elétrica, redes biológicas, em processo de agendamento. O método, como seu nome sugere, simula o processo biológico de seleção natural.

Mantém-se uma população de indivíduos cujas características são armazenadas em uma cadeia de bits de comprimento fixo, representando o genótipo biológico. A cada indivíduo se atribui uma pontuação de adaptação, dependendo da resposta dada ao problema por este indivíduo. Aos mais adaptados é dada uma maior oportunidade de se reproduzirem mediante cruzamentos com outros indivíduos da população, processo de *Crossover* (figura 2), produzindo descendentes com características de ambas as partes (LAZZUS e RIVERA, 2013).

Figura 2: Esquema de crossover e mutação.



Fonte: ABRANDAO. Genetic algorithms in PHP code example of (evolutionary programming).*Post*, 2015. Disponível em: <<http://www.abrandao.com/2015/01/21/simple-php-genetic-algorithm/>>. Acessado em: 04/03/2015.

De acordo com Sambridge e Drijkonigen (1991), complementando a exposição anterior, apenas indivíduos selecionados da população são permitidos formar descendentes. A seleção é baseada em aptidão: indivíduos com melhores valores de aptidão são selecionados mais frequentemente que indivíduos com piores valores.

A divisão dos genes dos pais é feita em lugares aleatórios. Porém, aqui deve-se fazer uma ressalva: separando a cadeia de *bits* em colunas, as colunas não podem se misturar, ou seja, coluna 1 do pai só pode ser transferida para coluna 1 do(s) descendente(s) e assim para as outras colunas.

Há também a operação de mutação que muda aleatoriamente um *bit* dentro da cadeia de *bits* do indivíduo, fazendo com que o espaço de solução seja mais explorado. Técnicas avançadas como “*niching*” (“de nicho”) e *elitism* (“de elite”) evitam que os indivíduos procurem em apenas uma parte do espaço e que os melhores indivíduos fiquem estagnados entre gerações, respectivamente (SHERER *et al.*, 2012).

Assim como acontece para o Método Enxame de Partículas, existem diversas adaptações para o Algoritmo Genético. Aqui foi apresentado um resumo do modelo padrão para ambos.

3. MATERIAIS E MÉTODOS

Considerando que o objetivo deste trabalho é de apresentar um algoritmo (algoritmo MEP) que utiliza meta-heurísticas para alterar parâmetros de um modelo para simulação visando otimizar este modelo, a pesquisa aqui desenvolvida caracteriza-se como exploratória, pois, segundo os autores Sampieri *et al.* (1991), ela pode ser utilizada quando o pesquisador buscar familiarizar-se e aprofundar a compreensão de um problema de pesquisa. Também pode ser utilizado este tipo de pesquisa para ajudar no desenvolvimento ou na criação de hipóteses explicativas de fatos a serem verificados em uma pesquisa causal. A pesquisa exploratória também é realizada quando se busca verificar se pesquisas semelhantes já foram

realizadas, quais os métodos utilizados e quais os resultados obtidos, determinar tendências, identificar relações potenciais entre variáveis e estabelecer rumos para investigações posteriores mais rigorosas. Então, para realizar a familiarização e aprofundamento do tema, inicialmente foi realizada uma pesquisa bibliográfica para buscar trabalhos de pesquisa semelhantes ao nosso objeto de estudo e também para identificar artigos que apresentassem dados que descrevessem um problema real a ser simulado.

Procurou-se por trabalhos que abordassem simulação e descrevessem um modelo de uma organização, mais especificamente de uma cadeia de processos, com certo grau de detalhamento (tempos, recursos, entre outros). Utilizar-se-á do modelo para aplicar o algoritmo MEP e analisar sua resposta, buscando otimizar este modelo frente a diferentes demandas/exigências. A análise se dará pela comparação dos valores de desempenho do modelo antes e depois da otimização.

4. RESULTADOS

Nesta seção apresenta-se o modelo para simulação, o algoritmo desenvolvido, a resposta a ser analisada pelo algoritmo após o processo de simulação, e o centro cirúrgico. Na sequência simula-se um aumento de demanda por cirurgias e ampliação dos recursos necessária para eliminar filas.

4.1 O modelo de simulação

O modelo do centro cirúrgico foi desenvolvido a partir da análise do trabalho “Modelagem e análise de um novo centro cirúrgico para um hospital em crescimento: uma abordagem baseada em simulação” dos autores Joaquim e Vieira (2009). O centro cirúrgico é do Hospital Universitário da PUCPR.

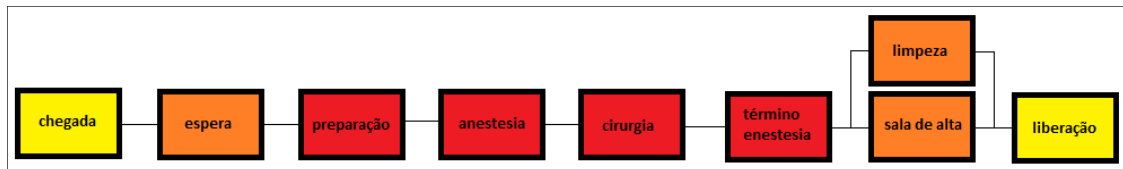
Neste trabalho, os autores têm como objetivo “avaliar alterações no Centro Cirúrgico com o intuito de se preparar melhor para o aumento de demanda esperado para os próximos anos” (JOAQUIM e VIEIRA, 2009). Para isso propõem quatro cenários diferentes, cada um com alterações no número de salas de cirurgia e uma reestruturação nas atividades internas do centro.

Como resultados, os autores chegaram à conclusão que o melhor cenário encontrado para atender bem a comunidade até 2010/2011 (estudo feito em 2009) deverá ter duas novas salas de cirurgia (totalizando 8) e uma reestruturação das atividades internas do centro. Isto reduzirá o tempo de espera do paciente por sala cirúrgica e também a própria taxa de ocupação das salas (JOAQUIM e VIEIRA, 2009).

Desta forma, foram utilizadas informações deste trabalho para construção do modelo do centro cirúrgico e neste aplicar o algoritmo MEP.

A figura 3 apresenta uma simplificação do modelo desenvolvido para simulação no software Rockwell Arena (2015).

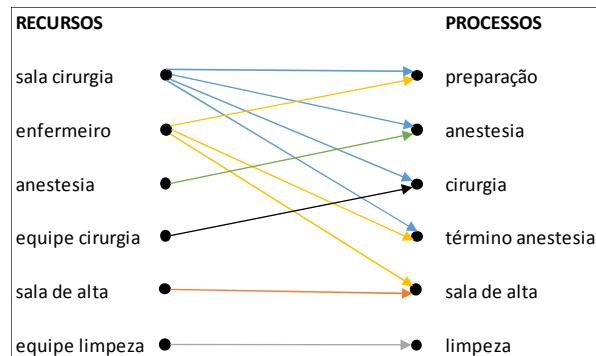
Figura 3: Esquema simplificado do modelo para simulação.



Fonte: Dos autores.

O modelo apresenta como características:

- Dois tipos de pacientes: de emergência e eletivo. Pacientes de emergência são os que não marcaram cirurgia e precisam de atendimento o mais breve possível, e eletivos aqueles que marcaram cirurgia (sem urgência). A chegada de pacientes eletivos ocorre com maior frequência.
- Cada paciente, após a sua chegada, passa a aguardar em uma fila a liberação de uma sala de cirurgia.
- Dá-se preferência para pacientes de emergência.
- Define-se o tipo de cirurgia, segundo uma distribuição pré-definida, e então o paciente é encaminhado para preparação da cirurgia.
- Há quatro especialidades de cirurgia: cirurgia geral, neurológica, ortopédica e outras. Todas possuem a mesma configuração de processos (processos em vermelho, figura 3), mas, em cada especialidade, esses processos possuem tempos de execução distintos.
- Os recursos utilizados são: sala cirurgia, equipe cirurgia, enfermeiro, anestesia, equipe limpeza e sala de alta. Esses recursos são compartilhados entre os processos, conforme mostra a figura 4.
- Procura-se sempre utilizar a capacidade máxima dos recursos, sem deixá-los ociosos e respeitando limites pré-definidos.
- Há mais de uma sala de cirurgia, ou seja, mais de uma cirurgia ocorre ao mesmo tempo.
- O processo de limpeza inicia quando o paciente passa para a sala de alta. Quando este processo de limpeza acaba, uma sala de cirurgia é liberada.

Figura 4: Distribuição dos recursos.

Fonte: Dos autores.

Neste trabalho, procura-se a otimização alterando apenas o número de recursos em cada processo sem alterar a sua cadeia (reestruturação das atividades), visto que esse seria um trabalho muito minucioso e não poderia ser feito pelo algoritmo MEP.

Salienta-se também que foi desenvolvido um modelo de centro cirúrgico mais próximo possível ao modelo real descrito no trabalho de Joaquim e Vieira (2009). Porém, pode haver divergências quanto ao modelo construído aqui e o modelo construído no trabalho citado, este também baseado no modelo real.

Dentre os recursos, pode-se definir a sala de cirurgia como recurso crítico, visto que é o recurso central, onde a cirurgia se inicia e termina, sendo assim é o mais utilizado. Outro fator que faz com que este recurso tenha um tratamento diferenciado é o custo envolvido tanto para a suposta instalação de novas salas de cirurgia quanto para manter as existentes.

4.2 O algoritmo MEP

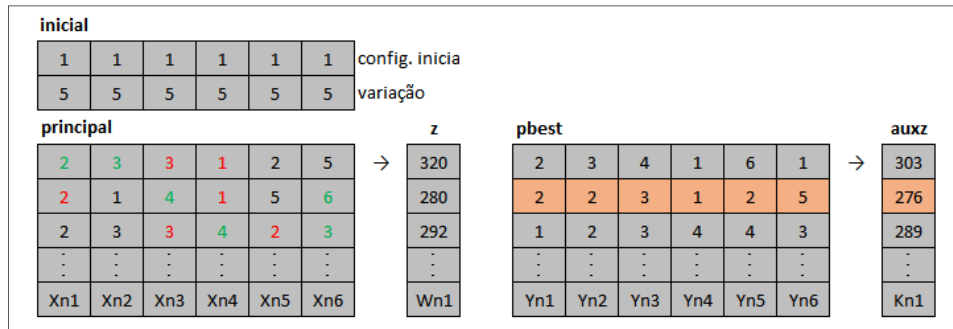
O algoritmo MEP foi desenvolvido em linguagem VBA dentro do software de simulação Rockwell Arena(2015). Ele utiliza fundamentos de duas meta-heurísticas, conhecidas como Método Enxame de Partículas e Algoritmo Genético, para encontrar a configuração de recursos que maximizem ou minimizem uma resposta do modelo, como por exemplo, tempos de espera, custos, produção e outros.

O usuário define a resposta a ser analisada, sendo possível escolher entre algumas pré-definidas. Também é possível utilizar expressões, combinando várias dessas “respostas”.

Cada configuração gerada pelo algoritmo MEP é lançada no modelo, sendo este simulado. Após a simulação é gerado um relatório com estatísticas do processo. É deste relatório que são extraídos os dados que a expressão solicita para formar o $z[X]$, sendo X a configuração em questão. Este valor é armazenado na matriz z .

A figura 5 mostra as principais matrizes utilizadas no algoritmo MEP (*inicial*, *principal*, *pbest*, *z* e *auxz*).

Figura 5: Principais matrizes utilizadas no algoritmo.



Fonte: Dos autores.

O funcionamento do algoritmo MEP segue os passos indicados a seguir.

1º O algoritmo extrai do modelo o número de processos e recursos para dimensionamento das matrizes. Extrai as capacidades, seus nomes e custos.

2º É solicitado um valor máximo de capacidades a ser aumentado em cada recurso.

3º A matriz *principal* é preenchida. São geradas novas configurações, respeitando os valores máximos para as capacidades. Cada linha apresenta uma nova configuração para o modelo.

4º Cada nova configuração gerada é implantada no modelo e, este então, simulado. No final é gerado o relatório, onde são extraídos os valores solicitados pela expressão. Calcula-se o $z[X]$ para cada configuração X preenchendo, desta forma, a matriz z .

$z[X] = \text{Custo Total} / \text{Produção}$, por exemplo.

A matriz *pbest* é preenchida, reportando as melhores configurações que passaram pela matriz *principal*.

5º Encontra-se então o *gbest*, linha com melhor valor entre todas as linhas aferidas (destacada em rosa na figura 5). Este valor se encontra na matriz *pbest*.

6º Neste ponto, inicia-se outra iteração. A matriz *principal* é preenchida com novos valores. Cada linha é então dividida em três setores (ver cor dos números na matriz *principal* figura 5), dos quais um receberá os números do *gbest* (números em vermelho), outro do *pbest* (números em verde) e o último será preenchido com números aleatórios (números em preto). A inserção de números aleatórios garante que seja explorada a maior variedade possível de configurações sem que o algoritmo se estagne, gerando configurações repetidas.

7º Com a nova matriz *principal*, todo o ciclo é repetido e cada nova configuração X é implantada no modelo, gerando após a simulação um novo $z[X]$.

8º Ao final do número de iterações definidas pelo usuário, o algoritmo MEP é finalizado e fornece a informação armazenada em *gbest*, a configuração de capacidades para otimização do modelo, ou seja, aquela configuração X que gerou o melhor $z[X]$.

9º É gerado um arquivo Excel onde são armazenadas as melhores configurações encontradas pelo algoritmo MEP no decorrer de sua execução, além de outras informações relevantes.

4.3 Resposta analisada

Nesta otimização, a resposta analisada é o **quociente** entre o **maior tempo de espera entre entidades (MAEE)** e a **menor utilização entre recursos (MEUR)**. A este quociente dá-se o nome de $z[X]$, onde X representa a configuração que, quando inserida no modelo da origem aos MAEE e MEUR. Assim, procura-se minimizar a resposta de forma a diminuir o tempo de espera e aumentar a utilização dos recursos. Se a espera diminui e a utilização aumenta, $z[X]$ diminui.

Os valores para o tempo de espera se encontram entre 0 e infinito, já para a utilização dos recursos encontra-se entre 0 e 100 (%).

4.4 O centro cirúrgico

No centro cirúrgico real, há 6 salas de cirurgia. Este número de salas é, em determinado momento, suficiente para suprir a demanda de cirurgias sem gerar filas significativas.

Nesta parte, é utilizado o algoritmo MEP para dimensionar/balancear os recursos do modelo de simulação tendo como condição inicial o recurso *sala de cirurgia* fixado em 6 unidades sem possibilidade de aumento. Aos demais recursos, não é definido valor máximo, sendo este definido pelo algoritmo MEP.

O relatório gerado, ao utilizar-se 10 partículas e 100 iterações, pode ser visto na figura 6.

Cada linha em “Configuração” representa uma configuração X para o modelo, sendo a amarela a inicial (anterior à otimização). As colunas em branco apresentam as respostas utilizadas para o cálculo dos valores de $z[X]$ (maior tempo de espera MAEE e menor utilização dos recursos MEUR, respectivamente) e, em vermelho, os valores de $z[X]$. Por fim, em verde, é apresentado em qual iteração a configuração foi formada.

A melhor configuração encontrada para o modelo é, portanto, [6, 5, 2, 1, 1, 7]. Isso mostra que, para o centro cirúrgico atender a demanda definida e com 6 salas de cirurgia, são necessários 5 enfermeiros, 2 anestesistas, 1 sala de alta, 1 equipe de limpeza e 7 equipes de cirurgia.

Na tabela 4 verifica-se o desempenho do sistema com a configuração de recursos estabelecida. *NumberIn* e *NumberOut* são, respectivamente, o número de pacientes que entraram no sistema (solicitando cirurgia) e o número de pacientes atendidos (cirurgia concluída e paciente liberado).

Figura 6: Configuração de recursos para o modelo.

Configuração					MAEE	MEUR	z	Iteração
1	1	1	1	1	3293,317	8,318	395,932	0
5	2	1	1	7	132,756	21,542	6,163	53
5	2	1	1	7	132,756	21,542	6,163	79
5	2	1	1	10	132,756	21,542	6,163	49
5	2	2	2	7	132,756	18,745	7,082	81
3	2	1	2	9	183,245	22,672	8,082	41
3	2	1	1	8	239,534	25,553	9,374	47

Tabela 4: Análise de desempenho da simulação com a configuração definida.

Maior espera registrada (min)		Média espera (min)		NumberIn (pacientes)	NumberOut (pacientes)	Menor ocupação recursos (%)
Eletiva	Emergência	Eletiva	Emergência	232	222	22
130,31	132,76	20,439	14,553			

Fonte: Dos autores.

Analisando os dados, percebe-se que o tempo de espera 133 min é um valor elevado. Porém, ressalta-se que esse valor é o máximo registrado de espera entre os pacientes em um período de 5 dias com 16h de funcionamento do centro cirúrgico por dia (80h úteis). A média de espera registrada (entre eletiva e emergência) foi de 17,5 min e a menor ocupação dos recursos de 22%.

4.5 Ampliação da demanda

Em um centro cirúrgico, com o passar dos anos, o número de cirurgias aumenta. Isso é uma tendência natural, onde um dos fatores é o aumento da população. Outro pode estar ligado à longevidade das pessoas, que vem crescendo junto com a necessidade de cirurgias, principalmente ortopédicas.

Desta forma, é simulado um aumento de demanda por cirurgias no modelo do centro cirúrgico. Para isso, reduz-se o intervalo entre a chegada de pacientes de acordo com um “fator de redução” (tabela 5). Posteriormente, analisa-se como o sistema reage à mudança.

Tabela 5: Análise de desempenho da simulação com diferentes aumentos da demanda.

	Fator de Redução		
	3/4	2/3	1/2
Tempo chegada eletiva (min)	3/4 TL	2/3 TL	1/2 TL
Tempo chegada emergência (min)	3/4 TM	2/3 TM	1/2 TM
NumberIn (pacientes)	265	294	380
NumberOut (pacientes)	257	282	284
Maior espera registrada (min)	244,45	511,99	3.057,75
Média espera geral (min)	26,877	68,561	201,359
Menor ocupação recursos (%)	23	29	29

Fonte: Dos autores.

TL = tempo de chegada eletiva inicial; TM = tempo chegada de emergência inicial.

Com fator de redução de 3/4, percebe-se que a espera máxima cresceu significativamente, 244 min, enquanto a média aumentou em aproximadamente 6 min, ficando em aproximadamente 27 min. A menor ocupação aumentou em apenas 1%, seguindo para 23%.

Outro dado importante a ser analisado tem relação com os parâmetros *NumberIn* e *NumberOut*. Sendo seus valores próximos, pode-se afirmar que o sistema está equilibrado, ou seja, a demanda por cirurgias está sendo atendida, apesar de haver filas.

Partindo para o segundo caso, com fator de redução 2/3, é notável o aumento no tempo de espera de modo geral. A maior espera registrada mais que dobrou quando comparada ao caso anterior (512 min) e o tempo médio de espera ficou em 68,561 min, o que é um número alarmante para um centro cirúrgico. Analisando o *NumberIn* e o *NumberOut*, tem-se um sistema parcialmente equilibrado, isto é, o sistema encontra-se estável a princípio, mas se a simulação continuasse além das 80h, haveria grande chance de desequilíbrio devido ao acúmulo de pacientes nas filas. Analisando a menor ocupação dos recursos tem-se um salto para 29%, o que é compreensível devido à maior carga de trabalho em que os recursos foram submetidos.

Por último, no terceiro caso, onde o fator de redução é igual a 1/2, a média de espera dos pacientes ficou em mais de 3h. Esta situação tem como consequência filas enormes, inadmissíveis para um centro cirúrgico. Analisando o *NumberIn* e o *NumberOut* percebe-se o total desequilíbrio do sistema.

4.6 Ampliação dos recursos

Uma forma de atender a um aumento de demanda por cirurgias é aumentar o número de recursos (médicos, salas de cirurgia, enfermeiros, entre outros). Desta forma, o algoritmo MEP será executado de forma a acrescentar recursos ao modelo. Ele deve encontrar a melhor

configuração de recursos para os casos com fator de redução $3/4$ e $2/3$, buscando diminuir o tempo de espera.

O número de salas de cirurgia, sendo este um recurso crítico por demandar altos custos e espaço para construção (ambos limitados), será aumentado em apenas 2 unidades, fixando-se 8 salas cirúrgicas. Os demais recursos não são limitados, sendo o algoritmo MEP a definir seus valores máximos.

Desta vez, estes recursos não iniciarão com valores unitários, mas sim, permanecerão com os seus valores atuais [6, 5, 2, 1, 1, 6]. Assim, partindo de um estado pré-otimizado, espera-se que o algoritmo MEP encontre a configuração “ótima” de forma mais rápida.

Segundo o relatório na figura 7, a melhor configuração para o modelo é [8, 7, 3, 3, 3, 8]. Isso mostra que, para este fator de redução e fixadas 8 salas de cirurgia, são necessários 7 enfermeiros, 3 anestesiastas, 3 salas de alta, 3 equipes de limpeza e 8 equipes de cirurgia. Na tabela 6 apresenta-se o desempenho do sistema com essa configuração.

Figura 7: Dimensionamento dos recursos com fator de redução de $3/4$.

Configuração					MAEE	MEUR	z	Iteração
5	2	1	1	6	244,446	23,477	10,412	0
7	3	3	3	8	19,252	15,609	1,233	60
7	3	3	3	10	19,252	15,609	1,233	83
9	3	3	3	11	19,252	15,258	1,262	35
9	3	3	3	10	19,252	15,258	1,262	45
9	3	3	3	8	19,252	15,258	1,262	48
9	3	3	3	10	19,252	15,258	1,262	50

Tabela 6: Análise de desempenho da simulação com a configuração definida.

Maior espera registrada (min)		Média espera (min)		NumberIn (pacientes)	NumberOut (pacientes)	Menor ocupação recursos (%)
Eletiva	Emergência	Eletiva	Emergência			
7,097	19,252	0,115	0,639	234	229	16

Fonte: Dos autores.

A seguir, serão mostrados os resultados para o modelo com fator de redução de $2/3$ (figura 8 e tabela 7).

Figura 8: Dimensionamento dos recursos com fator de redução de 2/3.

Configuração						MAEE	MEUR	z	Iteração
5	2	1	1	6		511,988	28,863	17,738	0
7	4	3	2	8		66,332	21,645	3,065	35
7	4	3	2	8		66,332	21,645	3,065	75
7	4	3	2	8		66,332	21,645	3,065	86
8	4	3	2	10		66,332	21,49	3,087	86
10	4	3	2	11		66,332	19,753	3,358	65
6	4	3	2	11		71,328	19,873	3,589	43

Tabela 7: Análise de desempenho da simulação com a configuração definida.

Maior espera registrada (min)		Média espera (min)		NumberIn (pacientes)	NumberOut (pacientes)	Menor ocupação recursos (%)
Eletiva	Emergência	Eletiva	Emergência			
66,332	38,995	3,812	2,382	329	322	22

Fonte: Dos autores.

A melhor configuração encontrada para o modelo neste caso é [8, 7, 4, 3, 2, 8]. Com este fator de redução e fixando 8 salas de cirurgia, são necessários 7 enfermeiros, 4 anestesistas, 3 salas de alta, 2 equipes de limpeza e 8 equipes de cirurgia.

Como visto anteriormente, para o caso onde o fator de redução é de 1/2 tem-se um sistema desequilibrado. Isto se deve à limitação imposta para salas de cirurgia (6 e depois 8) que não são suficientes para atender a demanda. Desta forma, é simulada uma última situação onde se definiu um número de 10 salas de cirurgia (figura 9). A configuração inicial foi mantida em [8, 7, 4, 3, 2, 8] (configuração ‘ótima’ da otimização com fator de redução 2/3).

A melhor configuração encontrada para o modelo neste caso é [10, 7, 6, 4, 4, 11]. São necessárias 10 salas de cirurgia, 7 enfermeiros, 6 anestesistas, 4 salas de alta, 4 equipes de limpeza e 11 equipes de cirurgia para atender a demanda. A tabela 8 mostra que com essa configuração é possível ter uma média de espera baixa e aceitável tanto para pacientes eletivos quanto para os de emergência.

Figura 9: Dimensionamento dos recursos com fator de redução de 1/2.

Configuração						MAEE	MEUR	z	Iteração
8	7	4	3	2	8	660,563	26,633	24,802	0
10	7	6	4	4	11	92,699	19,434	4,77	69
10	7	6	4	4	12	92,699	19,434	4,77	58
10	7	6	4	4	14	92,699	19,434	4,77	40
10	7	6	4	4	16	92,699	19,434	4,77	33
10	7	6	4	4	16	92,699	19,434	4,77	33

Tabela 8: Análise de desempenho da simulação com a configuração definida.

Maior espera registrada (min)		Média espera (min)		NumberIn (pacientes)	NumberOut (pacientes)	Menor ocupação recursos (%)
Eletiva	Emergência	Eletiva	Emergência			
92,699	62,032	5,059	2,579	467	457	19

Fonte: Dos autores.

Dentre todas as simulações realizadas, o ponto mais preocupante está no tempo máximo de espera encontrado para os pacientes de emergência. Este é um dos motivos para a análise da ‘maior espera registrada’ como resposta do modelo e deve ser sempre o menor possível. Mas esse raciocínio nem sempre é atendido e a dificuldade encontra-se na limitação de recursos imposta, muitas vezes, por motivos financeiros da organização em estudo.

5. CONCLUSÕES

Neste trabalho procurou-se demonstrar como métodos de otimização podem trazer benefícios para projetos de simulação, principalmente na área de gestão de processos e recursos. Foi demonstrado o potencial de otimização do algoritmo MEP frente a um problema de simulação dado por um modelo de um centro cirúrgico real.

Os resultados obtidos foram satisfatórios; tem-se por exemplo, uma redução de aproximadamente 90% no maior tempo de espera registrada entre os pacientes para os dois primeiros casos simulados. Este resultado poderia ter sido alcançado com o dimensionamento manual dos recursos, mas destaca-se que ele foi feito de forma automática (pelo algoritmo) e otimizada (buscando diminuir a ociosidade dos recursos). Desta forma, obteve-se uma distribuição de recursos de forma eficiente frente a diferentes demandas por cirurgias.

Conclui-se, portanto, que o algoritmo pode trazer benefício à gestão, não como uma ferramenta de decisão final, mas sim, como uma sugestão para início do dimensionamento de recursos ou como a confirmação para a decisão já tomada. Salienta-se também que o ambiente hospitalar é apenas um contexto para este trabalho visto que o algoritmo MEP pode trabalhar com qualquer modelo de simulação desenvolvido no *software* Arena.

Como trabalhos futuros, procura-se ampliar as possibilidades de otimização do algoritmo. Busca-se dimensionar recursos para atender pré-requisitos como, por exemplo, procurar a configuração de recursos que retorne um tempo máximo de espera de 20 min sem ociosidades. Outra possibilidade é de trabalhar com custos, fator esse que não foi desenvolvido neste trabalho devido à dificuldade de definir/estimar custos para os recursos utilizados. Também pretende-se desenvolver a versatilidade do algoritmo MEP fazendo com que este trabalhe com as mais diversas respostas e nos mais diferentes cenários.

REFERÊNCIAS

- BOTASSOLI, G. T.; ALBERTI, R. A.; FURTADO, J. C. Simulação computacional para otimização de filas em processos. *Revista Geintec*, São Cristóvão, n. 2, v. 5, p. 2121-2135, abr./jun., 2015.
- BOWDEN, R., HALL, J. Simulation Optimization Research and Development. *Winter Simulation Conference*, Washington DC, v. 2, p.1693-1698, dez. 1998.
- CAPPANERA, P., VISINTIN, F., BANDITORI, C. Comparing resource balancing criteria in master surgical scheduling: a combined optimisation-simulation approach. *Journal Production Economics*, 158, p. 179-196, 2014.
- CARVALHO, L. S. de. Modelagem e simulação: poderosa ferramenta para a otimização de operações logísticas. Disponível em: http://prosub1b.xpg.uol.com.br/cariboost_files/Modelagem_e_Simula_C3_83_C2_A7_C3_83_C2_A3o__Poderosa_Ferramenta_para_a_Otimiza_C3_83_C2_A7_C3_83_C.pdf. Acesso em: jun. 2015.
- CHWIF, L.; MEDINA, A. C. Modelagem e simulação de eventos discretos: *teoria e prática*. São Paulo: Campus - Grupo Elsevier, 2006.
- COSTA, F. M. da. *Construção de modelo de simulação de sistema puxado de produção para melhorias de eficiência*. Braga, 2011. 139 p. Tese (Mestrado), Departamento de Engenharia e Gestão Industrial, Universidade do Minho, 2011.
- EPITROPAKIS, M. G.; PLAGIANAKOS, V. P.; VRAHATIS, M. N.. Evolving cognitive and social experience in Particle Swarm Optimization through Differential Evolution. *Journal Information Sciences*, p. 50-92, 2012.
- ESMIN, A. A. A. *Estudo de Aplicação do Algoritmo de Otimização por Enxame de Partícula na Resolução de Problemas de Otimização Ligados ao SEP*. Itajubá, 2005. Tese (Doutorado), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Itajubá, 2005.
- JIANG, Y.; LI, X.; HUANG, C. Automatic calibration a hydrological model using a master-slave swarms shuffling evolution algorithm based on self-adaptive particle swarm optimization. *Journal Expert Systems with Applications*, p. 752-757, 2013.
- JOAQUIM, É. D.; VIEIRA, G. E. Modelagem e análise de um novo centro cirúrgico para um hospital em crescimento: uma abordagem baseada em simulação. *Revista Produção*, São Paulo, n. 2, v. 19, p. 274-291, maio/ago. 2009.
- LAZZ'US, J. A.; RIVERA, M. Gas-solid phase equilibrium of biosubstances by two biological algorithms. *Revista Mexicana de Física*, La Serena/Chile, nov./ dez. 2013.
- MACHADO, C. M. L. *et al.* Making decisions in the hospital's supply chain: the new look for its model of management. *Journal Business & Management Review*, United Kingdom, n. 5, v. 4, p. 55-63, jan. 2015.
- ROCKWELL ARENA: Software de simulação. Rockwell Automation. Versão 14.7; 2015; Trial Version. Download disponível em: <https://www.arenasimulation.com/simulation-software-download>. Acesso em: jun. 2015.

SAKURADA, N.; MIYAKE, D. I. Aplicação de simuladores de eventos discretos no processo de modelagem de sistemas de operações de serviços. *Revista Gestão e Produção*, São Carlos, n. 1, v. 16, p. 25-43, jan./mar. 2009.

SAMBRIDGE, M.; DRIJKONINGEN, G. Genetic algorithms in seismic waveform inversion. *Geophysical Journal International*, Oxford, n. 2, v. 109, p. 323-342, ago. 1991

SAMPIERI, R. H.; COLLADO, C. F.; LUCIO, P. B. *Metodología de la investigación*. Journal McGraw-Hill, México, 1991.

SHERER, E. A. *et al.* Application of a single-objective, hybrid genetic algorithm approach to pharmacokinetic model building. *Journal of Pharmacokinetics and Pharmacodynamics*, n. 4, v. 39, p. 393-414, ago. 2012.

SOUZA, G. D. de; CARVALHO, M. do S. M. V. de; LIBOREIRO, M. A. M. Gestão da cadeia de suprimentos integrada à tecnologia da informação. *Revista de Administração Pública*, Rio de Janeiro, n. 4, v. 40, jul./ago. 2006.